การทำภาษาไทยกราฟิกบน 16 บิต ด้วย TURBO BASIC

อาจหาญ สัตยารักษ์

ผู้เขียนหยุดเขียน CAI ในวิชาฟิสิกส์ไปเสียหลายฉบับ เพราะมัวใช้เวลาฝึกวิทยายุทธ 16 บิตเพื่อพัฒนาตนเองให้เข้ากับยุคใหม่ของคอมพิวเตอร์ แต่แน่นอนผู้เขียนจะยังคงยืนอยู่ ณ จุดเดิมของตนเอง เพื่อมุ่งพัฒนา CAI ในสาขาวิชาฟิสิกส์เหมือนเดิม

สิ่งแรกที่ผู้เขียนคำนึงเมื่อจะทำ CAI นั้นก็คือ จะทำบนซอฟต์แวร์ใดจึงจะทำได้ทั้งรูป (กราฟิก) และส่วนที่เป็น TEXT โดยคิดว่าในด้านกราฟิกนั้นต้องมีคำสั่งพื้นฐานที่มากับซอฟต์แวร์จึงพร้อมจะนำมาใช้ได้ทันทีโดยไม่ต้องสร้างกันใหม่ ส่วนด้าน Text นั้นมักไม่มีปัญหาถ้าเราไม่ทำภาษาไทย แต่เมื่อเราทำ CAI เพื่อเด็กไทย ดังนั้นเราจึงหลีกเลี่ยงเรื่องการแสดงอักษรภาษาไทยนี้ไม่ได้ เมื่อลองดูหลายๆซอฟต์แวร์แล้ว ผู้เขียนก็เลือกที่จะเขียน CAI บนซอฟต์แวร์ที่ชื่อว่า TRUBO BASIC (โปรแกรมในแผ่นจริงชื่อว่า TB เพราะโปรแกรมชื่อ TUBRO นั้นเป็นซอฟต์แวร์ของภาษาปาสคาล หรือ TURBO PASCAL)

โดยมีเหตุผลที่สำคัญ คือ

1. มีคำสั่งด้านกราฟิกมากพอสำหรับเขียนรูปพร้อมแล้ว เช่น

1.1 ลงจุดโดยใช้คำสั่ง

PEST (X,Y), สี

1.2 เขียนเส้นตรงได้โดยคำสั่ง

LINE (X1,Y1)-(X2,Y2) สี

1.3 เขียนรูปสี่เหลี่ยม (เฉพาะกรอบ) ได้โดยคำสั่ง

LINE (X1,Y1)-(X2,Y2) สี, BF

1.4 เขียนรูปสี่เหลี่ยมทึบได้โดยคำสั่ง

LINE (x1, y1) – (x2, y2) สี, BF

1.5 เขียนวงกลมหรือวงรีได้โดยคำสั่ง

CIRCLE (CX,CY), รัศมี, สี, มุมเริ่มต้น, มุมสิ้นสุด, อัตราส่วนของแกน

1.6 ระบายรูปให้ทึบได้โดยคำสั่ง

PAINT (CX,CY), สีระบาย, สีกรอบ

1.7 มีคำสั่งพิเศษในการเขียนรูปอีกแบบหนึ่ง ตามรูปที่ 1 (จะอธิบายเพิ่มเติมในช่วงต่อไป)

หมายเหตุ ข้อความภาษาไทยในคำสั่ง X,Y,X1,Y1,X2,XY,CX, CY หมายถึง ให้ค่าเป็นตัวเลข

2. สามารถเขียนโปรแกรมได้ทั้งแบบที่มี Line number (สำหรับคนที่ยังคิดการใช้ BASIC อยู่) หรือเขียนโดยไม่ต้องมี Line number แบบภาษาโครงสร้างทั้งหลายได้ แต่ในที่นี้ผู้เขียนจะเขียนแบบที่ไม่ต้องมี line number ซึ่งดีกว่าแบบแรกมาก

3. สามารถเขียนโปรแกรมแบบ Word Processor ได้ คือเลื่อนเคอร์เซอร์แบบ full screen, ค้นหาข้อความ, ค้นหาและเปลี่ยนข้อความ, ก็อบปี้ข้อความได้ทั้งแบบย้ายข้อความ หรือเก็บลงแผ่นดิสก์ หรือจากแผ่นดิสก์เข้าเครื่อง (กล่าวคือส่วนนี้ทำงานได้เท่ากับเทอร์โบปาสคาลทุกประการ)

4. ใช้กราฟิกได้ 3 แบบ คือ 320×200 จุด, 620×200 จุด และขนาด 620×350 จุด

5. สามารถเขียนชุดคำสั่งใหม่ (Procedure) ได้ ทำให้สามารถสร้างรูปภาพเก็บไว้ใช้ได้ (และยังสามารถนำไปใช้ในโปรแกรมอื่นๆได้ด้วย) ตัวอย่างของชุดคำสั่ง (Procedure) เช่น

```
SUB RING (x,y,r,C) ' เขียนรูปวงแหวน
    circle (x,y),r,c
    circle (x,y),r – 5,c
    paint (x + r – 2,y),c,c
END SUB
```

หมายเหตุ ชุดคำสั่งเหล่านี้จะเรียกใช้ได้โดยคำสั่ง CALL ซึ่งตามตัวอย่างก็คือใช้ CALL RING (100,80),50,2) เป็นต้น

และเหตุผลอื่นๆอีก ดังนั้น Turbo Basic จึงไม่ใช่ Basic แบบที่เรารู้จักกันมาก่อน ถ้าท่านลองมาศึกษาอย่างจริงจัง ผู้เขียนรับรองว่าท่านไม่เป็นคนเชยเลยที่ยังใช้ BASIC อยู่ รับรองว่าท่านจะภูมิใจที่ได้ใช้ซอฟต์แวร์ตัวนี้

สำหรับฉบับนี้ผู้เขียนขอเริ่มที่การเขียนตัวอักษรภาษาไทย ก่อน ทั้งนี้เพื่อจะได้เป็นเบื้องต้นของการเขียน CAI ครั้งต่อๆไปจะได้อ้างถึงได้โดยง่าย

เรียนรู้กับคำสั่งง่ายๆ

คำสั่งที่จะนำมาใช้ในการเขียนอักษรภาษาไทยคือ คำสั่งตามรูปที่ 1 นั่นเอง โดยใช้อักษรตามที่ระบุไว้ในรูปที่ 1 นั้นตามด้วยตัวเลขก็จะได้จุด หรือเคลื่อนที่โดยไม่ลงจุดได้ทันที (ซึ่งเป็นการเคลื่อนที่สัมพัทธ์) เช่น

U6 ก็คือเลื่อนขึ้น 6 จุดโดยลงจุดด้วย จะได้จุดทั้งหมด 7 จุด (รวมจุดเดิมด้วย)

m+3,−4 ก็คือเลื่อนไปทางขวา 3 จุด และขึ้นข้างบน 4 จุด (ลากเป็นเส้นทแยงมุม)

ถ้าเติม b นำหน้าก็จะไปโดยไม่ลงจุด คือ bu6, bm+3,−4

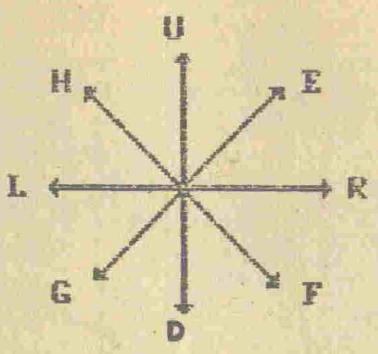
ถ้าเติม n นำหน้าก็จะลงจุดแล้วกลับมาที่เดิม คือ nu6,nm+3,−4

เฉพาะคำสั่ง m นั้นถ้าต่อด้วยตัวเลขที่ไม่มีเครื่องหมาย +, − จะเป็นการตรงไปยังจุดที่ระบุ เช่น m0,0 ก็คือตรงไปยังมุมบนซ้าย m20,50 คือไปยังจุดที่ 20,50 นั่นเอง

หมายเหตุ คำสั่งจริงต้องใช้ DRAW ร่วมด้วย คือ DRAW "u6"

DRAW "bu6" : DRAW "nm+3,−4" : DRAW "m+3,−4"

ผู้อ่านคงจะพอมองเห็นวิธีใช้คำสั่งเหล่านี้พอสมควร



Un วาดเส้นทิศทางขึ้น
Rn วาดเส้นทิศไปทางขวา
En วาดเส้นทิศตะวันออกเฉียงเหนือ
Hn วาดเส้นทิศตะวันตกเฉียงเหนือ
Dn วาดเส้นทิศทางลง
Ln วาดเส้นทิศไปทางซ้าย
Fn วาดเส้นทิศตะวันออกเฉียงใต้
Gn วาดเส้นทิศตะวันตกเฉียงใต้

นอกจากนี้มีรหัสเสริมคือ

M x,y ย้ายโดยกำหนดค่า x,y (ถ้ามีเครื่องหมาย + − จะเป็นการย้ายโดยนับจากจุดเดิม ถ้าไม่มีก็จะย้ายตรงไปจุดนั้นเลย)

b ใช้นำหน้าเมื่อต้องการย้ายโดยไม่ปรากฏจุด

TA, A ย้ายโดยระบุมุม (TA ได้ 0-360°, A ได้ 0-3°)

N ใช้นำหน้าในกรณีที่ต้องการลงจุดแล้วกลับมาจุดเดิม

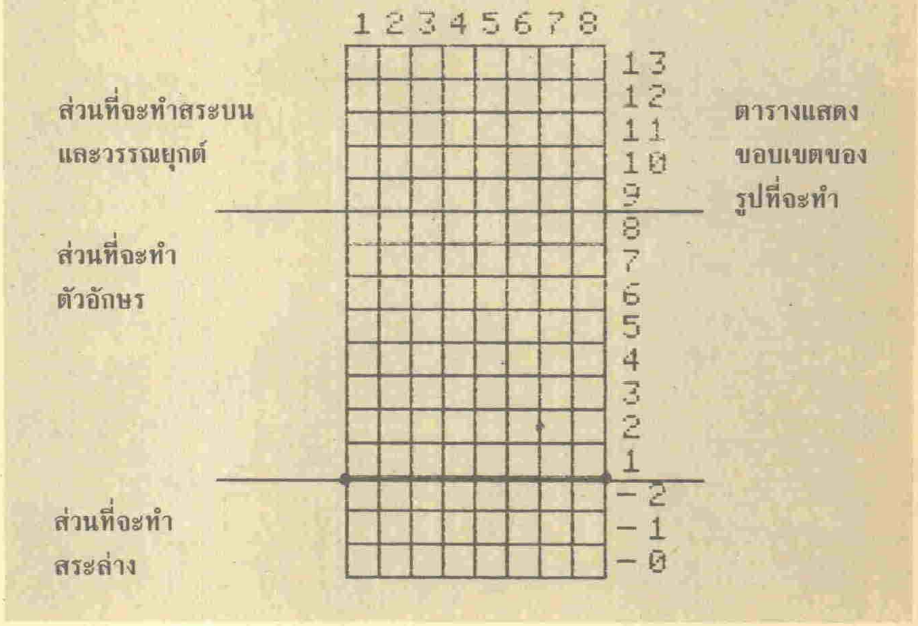
รูปที่ 1 แสดงรหัสกำหนดทิศทางของจุด

แนวคิดของการนำตัวอักษรภาษาไทย

ภาษาไทยเรามี 4 ระดับคือ ตัวอักษร สระล่าง สระบน และวรรณยุกต์ ดังนั้นเราจึงต้องวางรูปแบบของการทำดังตารางในรูปที่ 2 ซึ่งจะเป็นตารางขนาด 8×16 จุด โดยแบ่งเป็นส่วนๆดังนี้

แถว 1-8 สำหรับทำตัวอักษร โดยให้เริ่มที่จุด 1,1 จบลงที่จุด 8,1

แถว 9-13 สำหรับทำสระบนและวรรณยุกต์ โดยให้เริ่มที่จุด 8,1 จบลงที่จุด 8,1



รูปที่ 2 แสดงตารางขอบเขตของรูปที่จะทำ

แถว 0-2 (แถวล่าง) สำหรับทำสระล่าง โดยเริ่มที่จุด 8,1 จบลงที่จุด 8, 1 เราก็จะได้แนวการสร้างรูปอักษรซึ่งจะสามารถเขียนข้อความต่อกันได้ตามต้องการ

ลองมาดูตัวอย่างที่เราจะทำตัวอักษรภาษาไทยกันสักหน่อย สมมติว่าเราต้องการนำ ป กั ก่ เราอาจลงจุดได้ดังรูปที่ 3

ทีนี้มาลองดูการลง ป จะได้คำสั่งอย่างไร

1. เริ่มที่จุด 1, 1 (ยังไม่มีคำสั่ง)

2. ไปทางขวาโดยไม่ลงจุด 1 จุด และขึ้นข้างบนโดยไม่ลงจุด 6 จุด = bm+1, −6 (ตอนนี้มาอยู่ที่จุด 2, 7)

3. ไปทางขวาโดยลงจุด 1 จุด =r1 (ตอนนี้อยู่ที่จุด 3, 7)

4. ลงข้างล่างโดยลงจุด 6 ขุด =d6 (ตอนนี้อยู่ที่จุด 3, 1)

5. ไปทางขวาโดยลงจุด 4 จุด =r4 (ตอนนี้อยู่ที่จุด 7, 1)

6. ขึ้นไปข้างบนโดยลงจุด 7 จุด แล้วกลับที่เดิม = nu7 (ตอนนี้อยู่ที่จุด 7, 1)

7. ไปทางขวาโดยไม่ลงจุด 1 จุด =br1 (ตอนนี้อยู่ที่จุด 8, 1 แล้ว)

ดังนั้นรูปของ ป (สมมุติให้เป็น A$) คือ

A$ = "bm+1, −6r1d6r4nu7br1"

คราวนี้มาลองดู สระ, บ้าง

1. เริ่มที่จุด 8, 1

2. ไปทางซ้ายโดยไม่ลงจุด 3 จุด และลงข้างล่างโดยไม่ลงจุด 2 จุด = bm−3, +2 (ตอนนี้อยู่ที่จุด 5, −1)

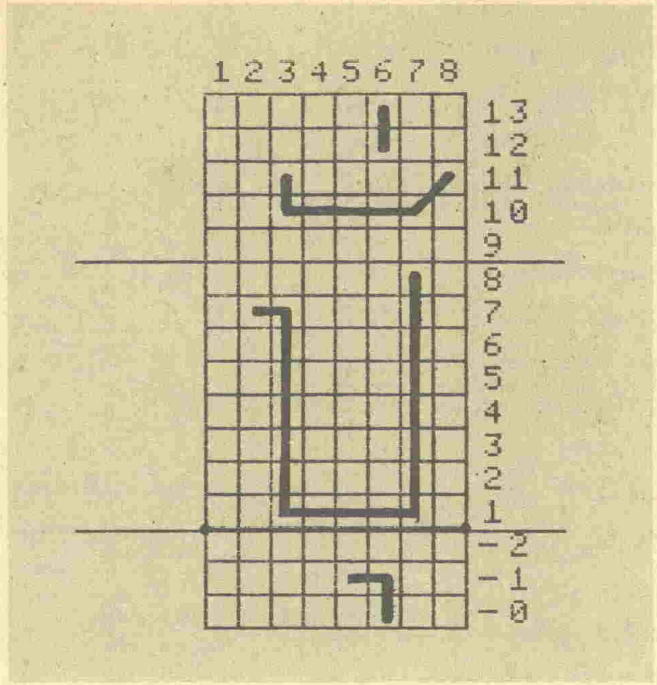
3. ไปทางขวาโดยลงจุด 1 จุด =r1

4. ลงข้างล่าง 1 จุด =d1

5. ไปทางขวาโดยไม่ลงจุด 2 จุด และขึ้นข้างบนโดยไม่ลงจุด 3 จุด = bm+2, −3 จึงมาอยู่ที่จุด 8, 1 ตามเดิม

ถ้ากำหนดเป็น B$ จะได้, ดังนี้

B$ = "bm−3, +2r1d1bm+2, −3"



รูปที่ 3 แสดงตัวอย่างการลงจุดของ ป ุ ั ่

การกำหนดตัวแปรมิติ (array) เพื่อให้สะดวกต่อการเขียนลงบนจอภาพ

การที่เราจะกำหนดเป็น A$, B$, C$ ไปเรื่อย ๆ นั้นคงไม่สะดวกต่อการเขียนโปรแกรมแน่ ๆ ดังนั้นเราจึงควรกำหนดให้เป็นตัวแปรมิติ (array) และเพื่อประหยัดเนื้อที่หน่วยความจำ เราก็ให้ตัวแรกคือ Space เป็นตัวแปรที่ 0 สมมุติว่าผู้เขียนเรียกใช้ตัวแปรว่า **Thai$** ก็จะใช้ค่า Thai$(0) = "br7" คือเมื่อพบว่าเป็น Space ให้ขยับไปทางขวาโดยไม่ลงจุด 7 จุด

ส่วนตัวอักษรอื่น ๆ นั้นก็หาจากค่า asc ขอบอักษรแต่ละตัวลบด้วย 32 เช่นอักษร ก คือตัว d ก็หาได้จาก print asc ("d") − 32 ซึ่งคือ Thai$ (68) อักษร ป คือตัว x ก็หาได้จาก print asc ("x") − 32 ซึ่งคือ Thai$ (88)

หรือจะหาค่าของแต่ละตัวอักษรได้ตามรูปที่ 4 (โปรแกรมที่ 1) โดยผู้เขียนถือว่าไม่ได้ทำบนการ์ดที่มีภาษาไทย (แต่ถ้าผู้อ่านมีการ์ดภาษาไทยก็คงจะเปลี่ยนแปลงค่าได้) ถ้าผู้อ่านไม่

```
cls
   a$ ="d-8'OC=:GPED{R<If95mTo[xz/ra4,pi1A;LKls>vV" 'ก-ฮ
     for k = 1 to len(a$)
        ? mid$(a$,k,1)"     " asc(mid$(a$,k,1))-32
           input "",b$
     next
```

รูปที่ 4 โปรแกรมที่ 1 การหาค่าตัวแปรมิติ (array ของ Thai$)

ต้องการตำแหน่งตัวอักษรตามโปรแกรมที่ 1 ก็กรุณาแก้ไขค่าใน A$ เองได้เช่นกัน

มาถึงตอนนี้ท่านก็จะได้แนวทางการทำตัวอักษรและค่าตัวแปรมิติของแต่ละตัวอักษรแล้ว จากนี้เราจะทำอย่างไรจึงจะแสดงอักษรภาษาไทยที่สร้างเสร็จแล้วนี้ให้ปรากฏบนจอภาพในขณะที่โปรแกรมทำงานได้

เรื่องนี้สามารถทำได้โดยกำหนดชุดคำสั่งขึ้นมา 1 ชุด ดังรูปที่ 5 (โปรแกรมที่ 2) ซึ่งตามโปรแกรมนี้ผู้เขียนไม่ได้กำหนดค่า Thai$ ไว้ให้ทั้งหมดนะ (ทั้งนี้เพื่อผู้อ่านได้ลองใช้ความสามารถของตนเองทำขึ้นมาตามแนวความคิดที่ให้ไว้นั่นเอง)

ชุดคำสั่ง (Procedure) นี้ผู้เขียนให้ชื่อว่า **Printthai** มีรูปแบบดังนี้

SUB Printthai (row, coloum, a$, cc) โดยมีตัวแปรคือ

row บรรทัดที่ต้องการจะเขียน

column คอลัมน์ที่ต้องการจะเขียน

A$ ข้อความที่ต้องการจะเขียน

cc สีที่ต้องการให้อักษรปรากฏ (ถ้าไม่ต้องการก็ให้ตัดออกได้)

โดยเป็นตัวแปรที่ผู้ใช้ต้องกำหนดก่อนเรียกชุดคำสั่งนี้ เช่น CALL Printthai (3, 6 "vk0skP", 2) เมื่อลองรันดูก็จะได้ข้อความที่บรรทัดที่ 3 แสดงไว้ในโปรแกรมแล้ว

นอกจากผู้อ่านจะใช้คำสั่ง DRAW ในลักษณะแล้ว ก็ยังใช้เขียนรูปอื่นได้อีกมากมาย ซึ่งแน่นอนเพื่อความสะดวกในการใช้งาน และเรียกใช้ในโปรแกรมอื่นๆ จึงควรสร้างเป็นชุดคำสั่ง (Procedure) เก็บไว้

ดังนั้นในแต่ละตอนที่ผู้เขียนจะพูดถึงบทความด้าน CAI วิชาฟิสิกส์ครั้งต่อ ๆ ไป จึงอาจอ้างถึงชุดคำสั่งเดิมที่เขียนไปในฉบับก่อน ๆ ได้ทันที

นั่นคือเมื่อเราต้องการชุดคำสั่งเดิมจากแผ่นดิสค์เราก็สามารถเรียกมาใช้ได้ 2 แบบ คือเรียกมาให้ปรากฏเลยด้วยการกด CTRL-K ตามด้วย r แล้วบอกชื่อโปรแกรมที่เก็บชุดคำสั่ง *(รายละเอียดเพิ่มเติมควรอ่านจากหนังสือ Turbo Basic Owner's Handbook ของบริษัท BORLAND INTERNATIONAL INC.)* หรือเรียกใช้ได้ด้วยคำสั่ง $INCLUDE ก็ได้

ผู้เขียนคิดว่าเพียงเท่านี้ผู้อ่านจะสามารถสร้างภาษาไทยเองไว้ใช้เองได้อย่างครบถ้วนแล้ว ดังนั้นถ้าบทความ CAI วิชาฟิสิกส์ของผู้เขียนอ้างถึง CALL Printthai ที่ใดก็หมายถึงการเรียกหาชุดคำสั่งชุดนี้นี่เอง ทั้งนี้หมายถึงชุดคำสั่งอื่น ๆ ด้วย หากมีการ CALL ชุดคำสั่งแต่ปรากฏว่าไม่มีในบทความครั้งนั้น ก็หมายถึงชุดคำสั่งที่มีในบทความเรื่องที่ผ่านมาในฉบับก่อนนั่นเอง

พบกันใหม่ในฉบับหน้าด้วยโปรแกรมการเรียนการสอนวิชาฟิสิกส์เช่นเดิม

```
B>

SUB Printthai(row,column,a$,cc)     'ชุดคำสั่งชื่อ Printthai
      dim Thai$(200)                'กำหนดให้เท่ากับค่าตัวแปรสูงสุด
    Thai$(0)   ="br7"                           'ช่องว่าง
    Thai$(68)  ="br1u3e1h1e1r3f1d5br1"          ' ก

       ใส่รหัสของตัวอักษรที่ท่านทำเป็นตัวแปรมิติในส่วนนี้
                ไปจนครบตัวอักษรที่ทำ

    Thai$(88)  ="bm+1,-6r1d6r4nu7br1"           ' บ
    Thai$(200) ="br6h2u2e2bd6br1"
  ' BEGIN : DISPLAY ON
     row =int(row*16)-4 : column = int((column-1)*8)    'หาตำแหน่งเริ่มต้น
       b$ ="bm" + str$(column) + "," + str$(row) : draw b$ ' ไปตำแหน่งเริ่มต้น
          col$ = "c" + str$(cc) : draw col$     'กำหนดสีของตัวอักษร
             DO
               k = k +1 :  n = asc (mid$(a$,k,1)) -32   'คำนวณหารูปตัวอักษร
                 draw Thai$(n)                  'วาดรูปอักษรที่คำนวณได้ลงบนจอภาพ
             LOOP UNTIL k = len(a$)             'วน loop จนกว่าค่า k เท่ากับ len(a$)
  ' END : DISPLAY OFF
END SUB
' ============================== MAIN PROGRAM ================================
      'ส่วนของโปรแกรมที่จะให้แสดงผลบนจอภาพ
    screen 1 : COLOR 16,0                       'เข้ากราฟิค และกำหนดสี
    CALL Printthai(2,3,"vk0skP ly9pkiydkN",2)   'รูปแบบของการให้แสดงข้อความ
' ============================ END OF PROGRAM ================================
```

รูปที่ 5 โปรแกรมที่ 2 แสดงชุดคำสั่งและการเรียกใช้